



# 구축 교육과정

교재

NIA 한국정보화진흥원



# PaaS-TA 구축



# PaaS-TA 배포를 위한 기본 구성

## » IaaS 종류

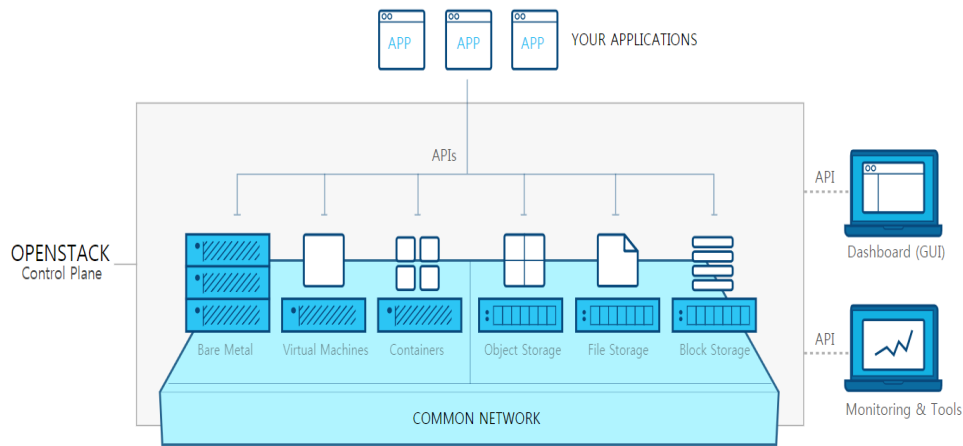


## PaaS-TA 배포를 위한 기본 구성

## » IaaS 종류



데이터 센터 전체에서 대규모 컴퓨팅, 스토리지 및  
네트워킹 리소스 풀을 제어하는 클라우드 운영 체제



관리자가 제어할 수 있도록 하는  
대시보드를 통해 관리됨

사용자가 웹 인터페이스를 통해  
리소스를 프로비저닝할 수 있도록 함

# PaaS-TA 배포를 위한 기본 구성

## » IaaS 종류



### 서버를 가상화하는 베어메탈 하이퍼바이저

: 호스트 운영체제에서 다수의 운영체제를 실행하기 위한 논리적 플랫폼

서버를 가상화하여 IT인프라 관리에 소모되는 시간과 비용을 절감

하드웨어를 많이 사용하지 않고도 애플리케이션을 통합할 수 있음

**ESXi 기반으로 개발** – 안정성과 성능, 그리고 환경 지원에 대한 업계 표준을 정립

**vSphere client 지원** – 간편하게 몇 분만에 가상 머신을 만들고 프로비저닝할 수 있음

**씬 프로비저닝을 지원** – 스토리지 리소스를 물리적인 스토리지의 실제 용량 이상으로 초과 할당할 수 있다는 의미. 메모리 리소스의 성능을 최적화

# PaaS-TA 배포를 위한 기본 구성

## » IaaS 종류



Amazon에서 제공하는 클라우드 서비스로,  
building blocks의 형태로 서비스를 제공

Building blocks은 클라우드에서  
모든 유형의 애플리케이션을 생성하고 배포하는 데 사용

광범위한 글로벌 클라우드 기반 제품을 제공



# PaaS-TA 배포를 위한 기본 구성

## » IaaS 종류



## Google Cloud Platform

구글 내부와 동일한 지원 인프라스트럭처 위에서  
호스팅을 제공하는 구글의 클라우드 컴퓨팅 서비스

GCP는 수십 개의 IaaS, PaaS 및 SaaS 서비스를 제공

GCP를 통해 인프라 관리, 서버 프로비저닝, 네트워크 구성으로  
발생하는 간접비에서 벗어날 수 있음

# PaaS-TA 배포를 위한 기본 구성

## » IaaS 종류



# Azure

글로벌 네트워크에서 구축, 관리 및 사용할 수 있는  
Microsoft의 클라우드 컴퓨팅 서비스

하이브리드 클라우드 환경을 통해 동일한 방식으로 응용 프로그램을 빌드 및 배포

클라우드 리소스를 모니터링하고 관리하고 보호하는  
통합형 개발 및 관리 도구를 통해 탁월한 생산성을 확보





MEMO



## BOSH 개념

**CLOUDFOUNDRY BOSH**

**Cloud Foundry BOSH**

### Stemcells

BOSH produces official [stemcells](#) for popular open source infrastructures that support it, [light stemcells](#) are also available and shared within the community.

Version	Download (SHA-256)
3586.25	Full Stemcell
3586.25	Full Stemcell
3586.25	Full Stemcell
3586.25	Light Stemcell

### Releases

A release is a versioned collection of configuration properties, configuration templates, start up scripts, source code, binary artifacts, and anything else required to build and deploy software in a reproducible way ([learn more](#)).

### Community Releases

- [DataDog/datadog-agent-boshrelease](#)
- [DataDog/datadog-firehose-nozzle-release](#)
- [Dynatrace/bosh-oneagent-release](#)
- [SAP/bosh-kubernetes-cpi-release](#)
- [SAP/ipsec-release](#)
- [bosh-packages/cf-cli-release](#)
- [cf-platform-eng/couchdb-boshrelease](#)
- [cf-platform-eng/docker-boshrelease](#)
- [cf-platform-eng/hadoop-boshrelease](#)
- [cf-platform-eng/mesos-boshrelease](#)

**Cloud Foundry**

Cloud Foundry Foundation active projects

Worldwide [https://www.cloudfoundry.org](#) [cf-dev@lists.cloudfoundry.org](#)

Repositories 396 People 308 Projects 0

Search repositories... Type: All Language: All

**cf-deployment**

The canonical open source deployment manifest for Cloud Foundry

Shell ★ 84 172 Apache-2.0 Updated 5 minutes ago

**relint-ci-pools**

★ 2 1 Apache-2.0 Updated 12 minutes ago

**garden-runc-release**

Shell ★ 45 43 Apache-2.0 Updated 18 minutes ago

BOSH는 PaaS를 이루는 **VM들을 설치 및 관리**하는 도구

## » BOSH

BOSH는 manifest 파일을 사용하여 사용자에게  
전체 시스템의 작동 방식과 각 부분의 상호 작용 방식을 정의하도록 요청

BOSH manifest는 해당 시스템의 모든 부분을 manifest를 통해 추적

시스템의 각 부분에 에이전트를 배치하여 하나의 제품에 오류가 발생하면 운영자에게  
신속하게 경고하고 문제를 자동으로 복구할 수 있음 (라이프 관리의 이점)

운영자는 일반적으로 BOSH 커뮤니티에서  
BOSH로 구축된 다양한 릴리스를 작성하고 관리하는 사람

## » BOSH

릴리스에서는 manifest 파일을 사용하여  
매번 동일한 방식으로 소프트웨어를 배포하는 데 필요한 모든 사항을 정의

manifest 파일은 소스 코드, 운영 체제 종속성(operating system dependencies)  
및 종속 파일을 명확히 식별 한 다음 모두 함께 패키지 함

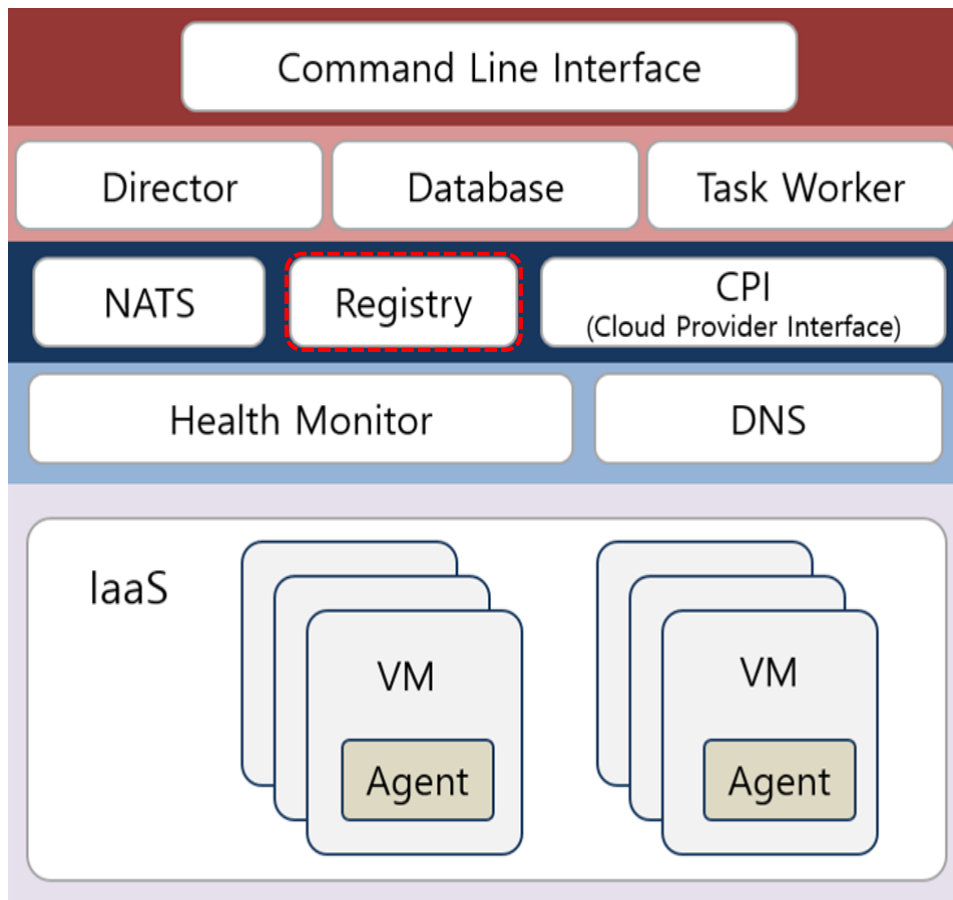
그 내용은 압축 된 폴더에 넣어 BOSH 서버에 저장됨

운영자가 해당 릴리스를 배포하려고 할 때마다 매번 동일한 방식으로  
릴리스를 예상대로 재구축하는 데 필요한 모든 것을 갖게 됨

소프트웨어를 전송하는 이러한 모든 방법은  
릴리스 엔지니어링(release engineering)이라는 개념에서 비롯됨

## » BOSH

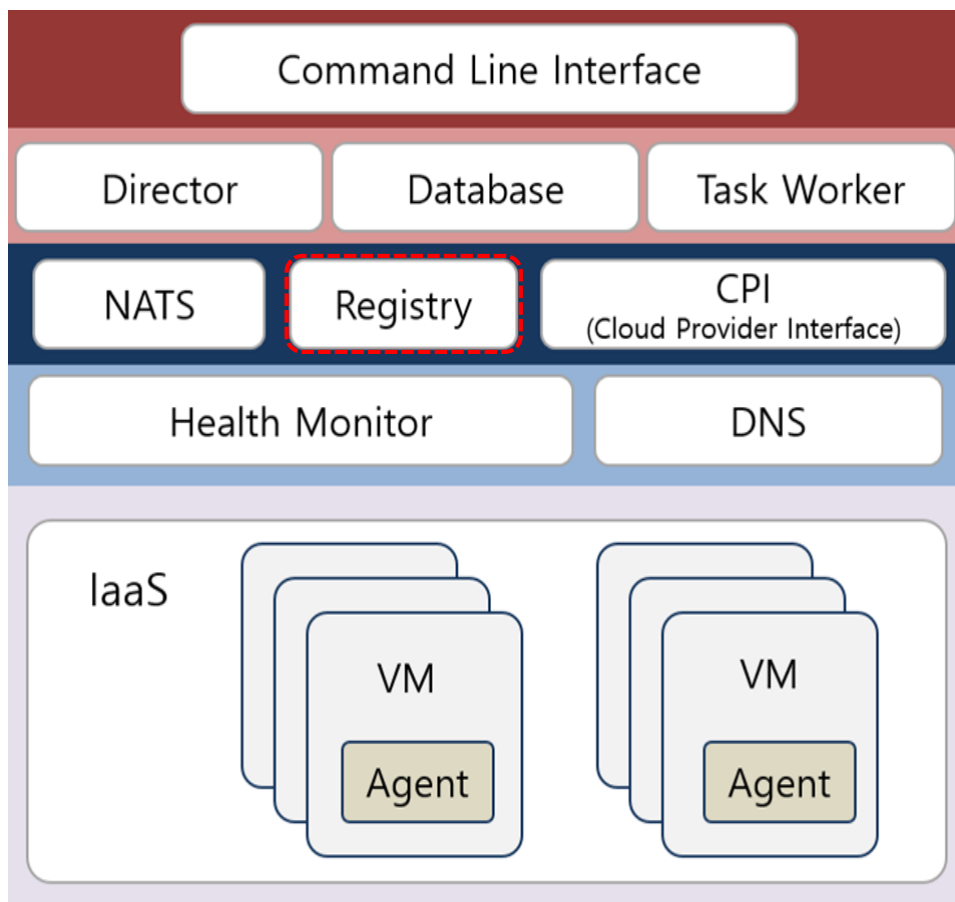
## Bosh 컴포넌트 구성



구성요소	설명
CLI	Director와 상호작용을 위한 CLI
Director	Director가 VM을 생성 또는 수정할 때 설정 정보를 레지스트리에 저장한다. 저장된 레지스트리 정보는 VM의 bootstrapping stage에서 이용된다.
NATS	Director 와 Agent 간의 메시징 시스템 VM에서 프로비저닝 지침을 수행하고 모니터링되는 프로세스의 상태 변화에 대해 Health Monitor에 알리는 역할을 수행한다
Registry	VM생성을 위한 설정정보 저장 * Not recommended
Health Monitor	Health Monitor는 BOSH Agent로부터 클라우드의 상태정보들을 수집한다. 클라우드로부터 특정 Alert이 발생하면 Resurrector를 하거나 Notification Plug-in을 통해 Alert Message를 전송할 수도 있다.

## » BOSH

## Bosh 컴포넌트 구성

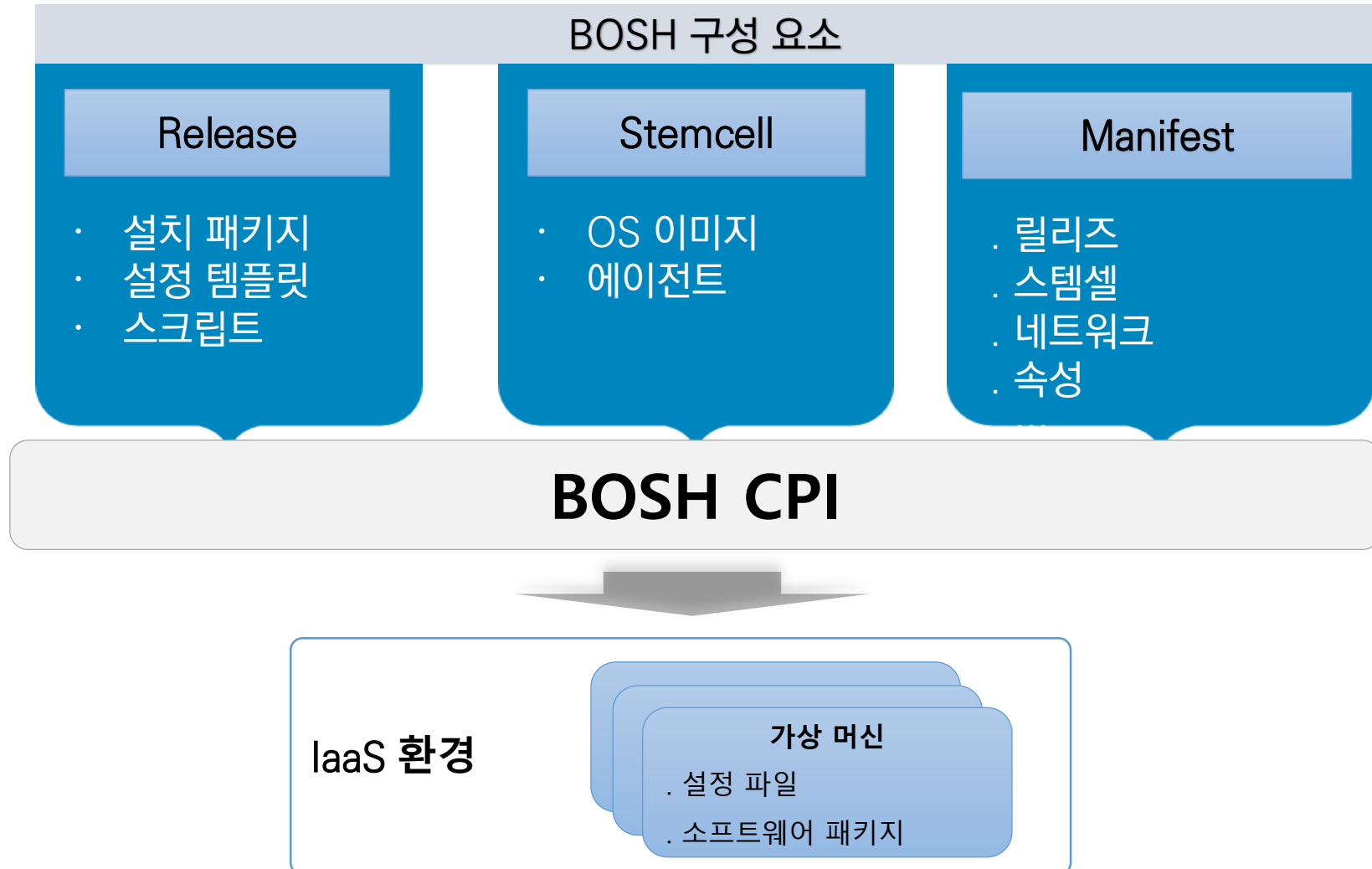


구성요소	설명
<u>Blobstore</u>	Release, compilation Package data를 저장하는 저장소이다
<u>UAA</u>	Bosh 사용자 인증 인가 처리를 한다.
Database	Director가 사용하는 데이터베이스로 Deployment시에 필요로하는 Stemcell / Release / Deployment의 메타 정보들을 저장한다.
Agent	VM에 설치되며, Director로부터 명령을 받아 개별 작업을 수행
DNS	배포된 VM의 DNS Resolution
Task Worker	Task Queue 로 부터 Task를 할당받아 수행

## 02

## BOSH 개념

## » BOSH





## 02

## BOSH 개념

## » Release

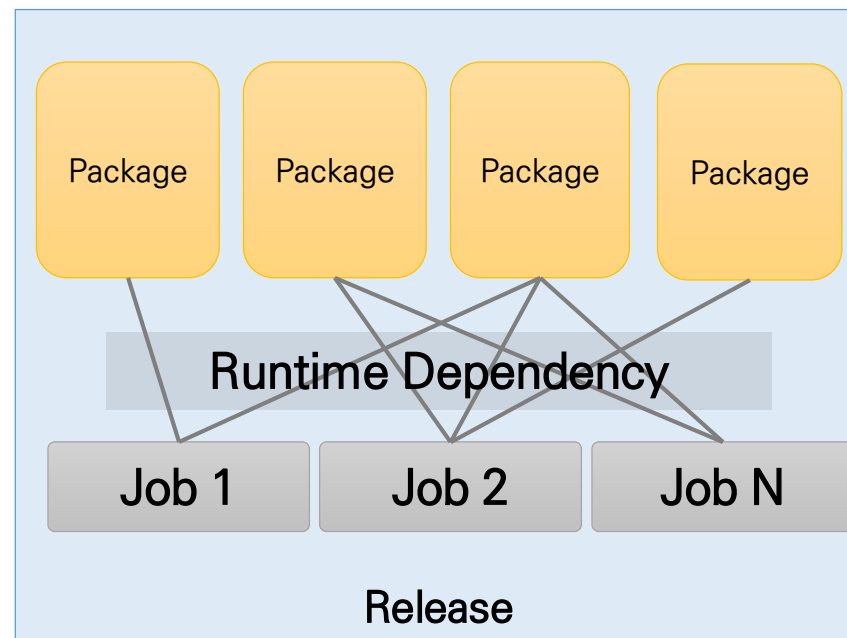
Packages – VM에 설치될 패키지

- 패키지
- 패키지 간 의존관계
- 설치 스크립트

Jobs – VM에서 수행될 프로세스

- 템플릿 : 배포 시 적용되는 설정 템플릿 파일
- 프로세스 시작 / 종료 스크립트

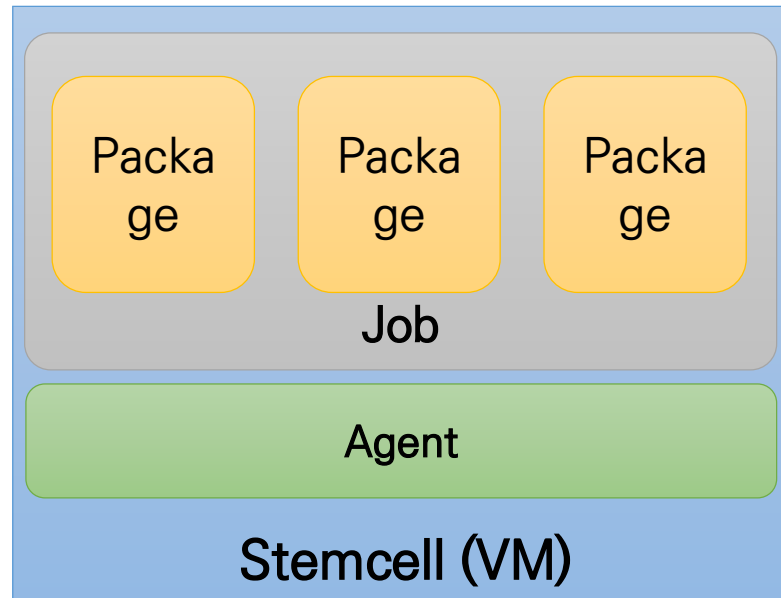
Release = Packages와 Job의 모음



## 02

## BOSH 개념

## » Stemcell



Stemcell 구성 : 기본 OS 이미지 + BOSH Agent

- Stemcell 자체는 서비스를 제공하지 않음
- Agent를 이용하여 Blobstore로 부터 Job 배포

## BOSH 개념

### » Manifest

#### 배포 Manifest

- BOSH Deployment manifest 는 components 요소 및 배포의 속성을 정의한 YAML 파일이다. Deployment manifest 에는 software를 설치 하기 위해서 어떤 Stemcell (OS, BOSH agent) 을 사용할 것이며 Release (Software packages, Config templates, Scripts) 이름과 버전, VMs 용량, Jobs params 등을 정의하여 BOSH deploy CLI 을 이용하여 software(여기서는 서비스팩)를 설치 한다.
- BOSH Deployment manifest 의 내용은 아래와 같다.
  - Deployment Identification: 배포 이름과 배포를 관리하는 BOSH Director의 UUID 정보
  - Releases Block: deployment 안의 각 release 의 이름 및 버전 정보
  - Stemcell Block: deployment 안의 각 stemcell 의 이름 및 버전 정보
  - Update Block: BOSH 가 배포 중에 작업 인스턴스를 업데이트 하는 방법을 정의
  - Instance Groups Block: Instance Groups 에 대한 구성 및 자원 정보
  - Properties Block: 글로벌 속성과 일반화된 구성 정보를 설명
  - Variables Block: Variables 정보
  - Networks Block: 네트워크 구성 정보
  - Compilation Block: 컴파일 시 필요한 가상 머신의 속성
  - Resource Pools Block: BOSH 로 생성하고 관리하는 가상 머신의 속성

## 02

## BOSH 개념

## » BOSH

## PaaS-TA 버전별 BOSH 구성

## PaaS-TA 3.1

PaaS-TA Container,  
Controller를 별도로  
deployment로 설치

## PaaS-TA 3.5 이상

PaaS-TA deployment  
**하나로 통합** 되었으며,  
**한번에 PaaS-TA를 설치**  
할 수 있음

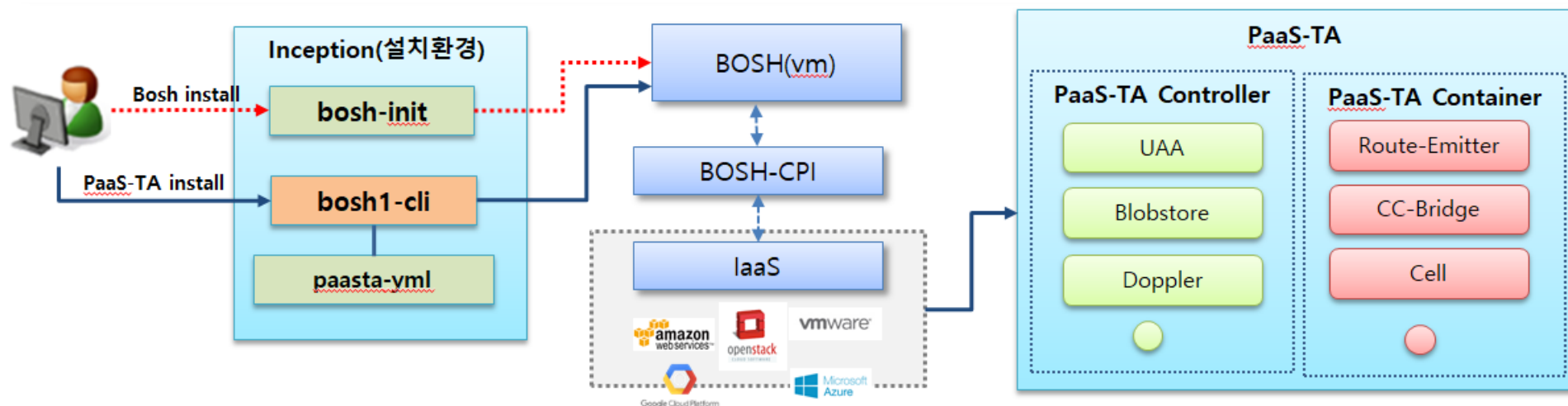
PaaS-TA 3.1까지는 cloud-foundry BOSH1을 기준으로 BOSH 설치

PaaS-TA 3.5부터는 BOSH2를 기준으로 Bosh 설치

BOSH2는 cloud-foundry에서 제공하는 bosh-deployment를 이용하여 BOSH 설치

## BOSH 개념

## » BOSH1 vs BOSH2

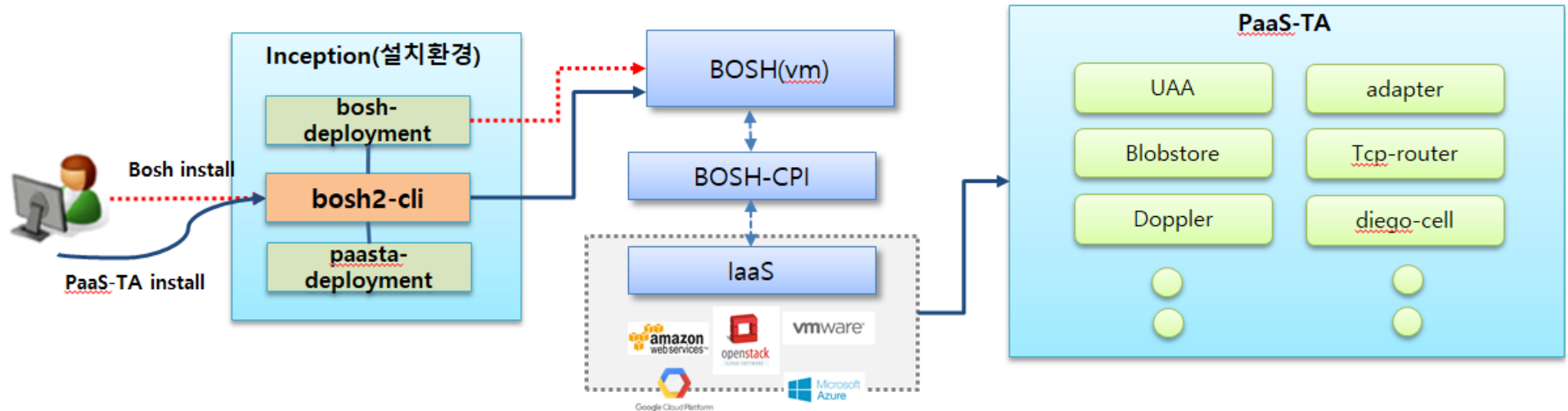


BOSH1은 bosh-init을 통하여 BOSH를 생성하고,  
BOSH1 cli를 통하여 PaaS-TA Controller, Container를 생성함

## 02

## BOSH 개념

## » BOSH1 vs BOSH2



BOSH2는 BOSH2-cli를 통하여 BOSH와 PaaS-TA 를 모두 생성 시킴

BOSH생성시 bosh-deployment를 이용하여 BOSH를 생성

BOSH생성 후 paasta-deployment를 활용하여 PaaS-TA를 생성

PaaS-TA-3.1 버전까지는 PaaS-TA Container, Controller를 별도로 deployment로 설치 해야 했지만 4.0(실제로는 3.5 이후)부터는 paasta-deployment 하나로 통합 되었으며, 한번에 PaaS-TA를 설치 할 수 있음



MEMO





## Director 설치

### » 샘플환경 설명 및 배포

#### Inception 서버 구성

PaaS-TA를 설치하기 위해 필요한 패키지 및 라이브러리,  
Manifest 파일 등의 환경을 가지고 있는 배포 작업 및 실행 서버

Inception 서버는 BOSH 설치와 BOSH의 Director를 설정

환경 구성에 있어서 전제조건으로 Inception 서버는 외부와 통신이 가능해야 함

BOSH 및 PaaS-TA 설치를 위해 Inception 서버에 구성해야 할 컴포넌트

- ✓ BOSH Cli 6.1 이상
- ✓ BOSH Dependency : ruby, ruby-dev, openssl 등
- ✓ BOSH Deployment: BOSH 설치하기 위한 manifest Deployment (<https://github.com/cloudfoundry/bosh-deployment> 기준)
- ✓ paasta deployment : PaaS-TA를 설치하기 위한 manifest deployment (PaaS-TA 5.0, cf-deployment 9.5 기준)

## 03

## Director 설치

## » 샘플환경 설명 및 배포

## Inception 설치

## Pre-requisite

본 설치 가이드는 Ubuntu 18.04 버전을 기준으로 함

Release, deployment 파일은 /home/{user\_name}/workspace/paasta-5.0  
이하에 다운로드 받아야 함

Inception machine spec. at least 8GB RAM and 100GB free disk space

## 03

## Director 설치

### » 샘플환경 설명 및 배포

#### Inception 설치

#### BOSH cli 및 dependency 설치

##### BOSH cli 설치

```
$ sudo apt-get update
$ mkdir workspace
$ cd workspace
$ sudo apt-get install -y curl
$ curl -Lo ./bosh https://s3.amazonaws.com/bosh-cli-artifacts/bosh-
cli-6.1.0-linux-amd64
$ chmod +x ./bosh
$ sudo mv ./bosh /usr/local/bin/bosh
$ bosh -v
```

## 03

## Director 설치

## » 샘플환경 설명 및 배포

## Inception 설치

## BOSH cli 및 dependency 설치

## BOSH dependency 설치

```
$ sudo apt-get install -y build-essential zlibc zlib1g-dev ruby ruby-dev  
openssl libxslt1-dev libxml2-dev libssl-dev libreadline7 libreadline-dev  
libyaml-dev libsqlite3-dev sqlite3
```

BOSH2 cli는 **BOSH deploy** 시 **BOSH certificate** 정보를 생성해 주는 기능이 있음

## 03

## Director 설치

## » 샘플환경 설명 및 배포

## Inception 설치

## Deployment 및 release 파일 다운로드

1

다운로드 파일이 위치할 경로 디렉토리를 생성

설치 파일 다운로드 : <https://paas-ta.kr/download/package>

```
$ mkdir -p ~/workspace/paasta-5.0/deployment
```

```
$ mkdir -p ~/workspace/paasta-5.0/release
```

```
$ mkdir -p ~/workspace/paasta-5.0/stemcell
```

## 03

## Director 설치

## » 샘플환경 설명 및 배포

## Inception 설치

## Deployment 및 release 파일 다운로드

2

다운로드 URL에서 [PaaSTA-Deployment] 파일을 다운로드 받아  
~/workspace/paasta-5.0/deployment 이하 디렉토리에 압축 해제

3

다운로드 URL에서 [PaaSTA-Release] 파일을 다운로드  
받아 ~/workspace/paasta-5.0/release 이하 디렉토리에 압축 해제

4

다운로드 URL에서 [PaaSTA-Stemcell] 파일을 다운로드 받아  
~/workspace/paasta-5.0/stemcell 이하 디렉토리에 압축 해제

## 03

## Director 설치

## » 샘플환경 설명 및 배포

## Inception 설치

## BOSH 환경 설정 및 디렉토리 설명

다운로드 받은 파일이 아래 경로에 존재하는지 확인

## BOSH dependency 설치

deployment	paasta-deployment, bosh-deployment, cloud-config가 존재
release	PaaS-TA Release File이 존재
stemcell	laaS별 Stemcell 이 존재



## Director 설치

### » 샘플환경 설명 및 배포

#### Inception 설치

#### BOSH 환경 설정 및 디렉토리 설명

paasta-5.0/deployment 이하 디렉토리

bosh-deployment	BOSH 설치할 manifest 및 설치 파일이 존재
cloud-config	PaaS-TA 설치하기 위한 IaaS 관련 network/storage/vm 관련 설정들을 정의 IaaS/network/disk 등 상황에 따라 설정이 다름
paasta-deployment	PaaS-TA 설치할 manifest 및 설치 파일이 존재

## 03

## Director 설치

## » 샘플환경 설명 및 배포

## Inception 설치

## BOSH 환경 설정 및 디렉토리 설명

paasta-5.0/release 이하 디렉토리

bosh	BOSH 설치 시 필요한 release 파일이 존재하는 디렉토리
paasta	PaaS-TA 설치 시 필요한 release 파일이 존재하는 디렉토리

paasta-5.0/stemcell 이하 디렉토리

paasta	PaaS-TA가 설치될 때 사용할 stemcell (paasta vm image)
--------	--------------------------------------------------

## 03

## Director 설치

## » 샘플환경 설명 및 배포

## Inception 설치

## BOSH 환경 설정

bosh-deployment 디렉토리로 이동

```
$ cd ~/workspace/paasta-5.0/deployment/bosh-deployment  
$ chmod 755 *.sh
```

~/workspace/paasta-5.0/deployment/bosh-deployment  
이하 디렉토리에는 IaaS별 BOSH를 설치 하는 shell이 존재

Shell 파일을 이용하여 BOSH를 설치

파일명은 `deploy-{iaas-name}.sh`로 만들어짐

## 03

## Director 설치

## » 샘플환경 설명 및 배포

## Inception 설치

## BOSH 환경 설정

deploy-aws.sh	amazon 설치 BOSH 설치 shell
deploy-openstack.sh	openstack 설치 BOSH 설치 shell
deploy-vsphere.sh	vsphere 설치 BOSH 설치 shell
deploy-gcp.sh	GCP(google) 설치 BOSH 설치 shell
deploy-azure.sh	Azure 설치 BOSH 설치 shell
deploy-bosh-lite.sh	BOSH-lite(local Test용) 설치 BOSH 설치 shell

## Director 설치

### » 샘플환경 설명 및 배포

#### Inception 설치

#### BOSH 환경 설정

설치 shell 파일은 각 IaaS별로 존재, BOSH 설치 시 명령어는 `create-env`로 시작

Shell이 아닌 BOSH command로 실행이 가능하며  
설치하는 IaaS 환경에 따라 Option들이 달라짐

<code>--state</code>	BOSH 설치 명령어를 실행할 때 생성되는 file로 설치된 BOSH의 IaaS 설정 정보를 보관 (backup 필요)
<code>--vars-store</code>	BOSH 설치 명령어를 실행할 때 생성되는 file로 BOSH가 설치 될 때 BOSH cli는 Bosh 내부 컴포넌트가 사용하는 인증서 및 인증정보를 생성 저장 (backup 필요)
<code>-o</code>	BOSH 설치 시 Option File을 설정할 수 있는데 IaaS별 CPI 선택 또는 jumpbox, credhub 등 설정 적용을 할 수 있음
<code>-v</code>	BOSH 설치 시 사용되는 yml 파일 또는 option 파일에 변수에 값을 설정할 경우 사용 할 수 있음 yml, optionfile 속성에 따라 필수인 경우와 option인 경우가 있음
<code>--var-file</code>	주로 인증서를 사용하는 경우 사용하는 option

## Director 설치

### » 샘플환경 설명 및 배포

#### Inception 설치

#### OPENSTACK BOSH 환경 설정

```
bosh create-env bosh.yml \
  --state=openstack/state.json \           # BOSH 설치 시 생성되는 파일로 절대 삭제 되면 안됨 (backup 필요)
  --vars-store=openstack/creds.yml \       # BOSH 내부 인증서 파일 중요 (Backup필요)
  -o openstack/cpi.yml \                   # openstack cpi 적용
  -o uaa.yml \
  -o credhub.yml \
  -o jumpbox-user.yml \
  -v inception_os_user_name='ubuntu' \
  -v internal_cidr=10.20.0.0/24 \           # internal ip range
  -v internal_gw=10.20.0.1 \               # internal ip gateway
  -v internal_ip=10.20.0.6 \               # internal ip
  -v director_name='micro-bosh' \         # BOSH director 명
  -v auth_url=http://xxx.xxx.xxx.xxx:5000/v3/ \ # keystone url
  -v az='nova' \                           # BOSH 설치될 az zone
  -v default_key_name='paasta-v50-key' \   # openstack key name
  -v default_security_groups=[paasta-v50-security] \ # openstack security group
  -v net_id=51b96a68-aded-4e73-aa44-f44a812b9b30 \ # network id
  -v multizone=true \                      # Compute Node Multizone 설정 여부
  -v openstack_password=xxxx \            # openstack user password
  -v openstack_username=xxxx \            # openstack user name
  -v openstack_domain='default' \         # BOSH 설치될 openstack domain name
  -v openstack_project='paasta' \         # BOSH 설치될 openstack project
  -v private_key=~/.ssh/paasta-v50-key.pem \ # openstack 접속 pem file
  -v region=RegionOne                     # BOSH 설치될 openstack 설치 될 region
```

## Director 설치

### » 샘플환경 설명 및 배포

#### Inception 설치

#### AWS BOSH 환경 설정

```
bosh create-env bosh.yml \
  --state=aws/state.json \           #BOSH 설치 시 생성되는 파일로 절대 삭제 되면 안됨 (backup 필요)
  --vars-store aws/creds.yml \       # BOSH 내부 인증서 파일 중요 (Backup필요)
  -o aws/cpi.yml \                   # aws cpi 적용
  -o uaa.yml \
  -o credhub.yml \
  -o jumpbox-user.yml \
  -v inception_os_user_name='ubuntu' \
  -v internal_cidr=10.0.0.0/24 \      # internal ip range
  -v internal_gw=10.0.0.1 \          # internal ip gateway
  -v internal_ip=10.0.0.6 \          # internal ip
  -v director_name='micro-bosh' \    # BOSH director 명
  -v access_key_id=xxxxx \           # aws access_key
  -v secret_access_key=xxxxx \       # aws secret_key
  -v region=ap-northeast-1 \         # BOSH가 설치될 aws region
  -v az=ap-northeast-1a \            # BOSH가 설치될 aws availability zone
  -v default_key_name='aws-paasta-rnd-v50-inception' \    # aws key name
  -v default_security_groups=[paasta-v50-security] \     # aws security-group
  -v private_key=~/.ssh/aws-paasta-rnd-v50-inception.pem \ # aws 접속 pem file
  -v subnet_id=subnet-ba1e15f3       # BOSH가 설치될 aws subnet
```



## Director 설치

### » 샘플환경 설명 및 배포

#### Inception 설치

#### VSPHERE BOSH 환경 설정

```

bosh create-env bosh.yml \
  --state=vsphere/state.json \      # BOSH 설치 시 생성되는 파일로 절대 삭제 되면 안됨 (backup 필요)
  --vars-store=vsphere/creds.yml \  # BOSH 내부 인증서 파일 중요 (Backup필요)
  -o vsphere/cpi.yml \              # vsphere 적용
  -o uaa.yml \
  -o credhub.yml \
  -o jumpbox-user.yml \
  -o vsphere/resource-pool.yml \      # vsphere 적용
  -v inception_os_user_name='ubuntu' \
  -v director_name='micro-bosh' \    # micro-bosh director 이름
  -v internal_cidr=10.30.0.0/16 \      # internal ip range
  -v internal_gw=10.30.20.23 \        # internal ip gateway
  -v internal_ip=10.30.40.111 \       # internal ip
  -v network_name='Internal' \       # internal 네트워크 이름 (vcenter)
  -v vcenter_dc=BD-DC \              # vcenter datacenter 이름
  -v vcenter_ds=iSCSI-28-Storage \    # vcenter data storage 이름
  -v vcenter_ip=10.30.20.22 \        # vcenter internal ip
  -v vcenter_user=administrator \     # vcenter user 이름
  -v vcenter_password=sdfsfsdeee \   # vcenter user 패스워드
  -v vcenter_templates=CF_BOSH2_Templates \ # vcenter templates 폴더명
  -v vcenter_vms=CF_BOSH2_VMs \      # vcenter vms 이름
  -v vcenter_disks=CF_BOSH2_Disks \   # vcenter disk 이름
  -v vcenter_cluster=BD-HA \         # vcenter cluster 이름
  -v vcenter_rp=CF_BOSH2_Pool        # vcenter resource pool 이름

```

## Director 설치

### » 샘플환경 설명 및 배포

#### Inception 설치

#### AZURE BOSH 환경 설정

```
bosh create-env bosh.yml \
  --state=azure/state.json \      # BOSH 설치 시 생성되는 파일로 절대 삭제 되면 안됨 (backup 필요)
  --vars-store azure/creds.yml \  # BOSH 내부 인증서 파일 중요 (Backup필요)
  -o azure/cpi.yml \              # azure CPI 적용
  -o uaa.yml \
  -o credhub.yml \
  -o jumpbox-user.yml \
  -v inception_os_user_name='ubuntu' \
  -v internal_cidr=10.0.0.0/24 \   # internal ip range
  -v internal_gw=10.0.0.1 \       # internal ip gateway
  -v internal_ip=10.0.0.6 \       # internal ip
  -v director_name='micro-bosh' \ # BOSH director 명
  -v vnet_name=paasta-net \       # Azure VNet 명
  -v subnet_name=bosh-net \      # Azure VNet Subnet 명
  -v subscription_id=816-91e9-4ba6-806c2ccb8630 \ # Azure Subscription Id
  -v tenant_id=aeacdca2-4f9e-8c8b-b0403fbdcfd1 \ # Azure Tenant Id
  -v client_id=779dae-49b05-464d3b877b7b \      # Azure Client Id
  -v client_secret='client-secret' \             # Azure Client Secret
  -v resource_group_name='paasta-v50-bosh-res-group' \ # Azure Resource Group
  -v storage_account_name='paastav50boshstore' \   # Azure Storage Account
  -v default_security_group='paasta-v50-nsg'        # Azure Security Group
```

## Director 설치

### » 샘플환경 설명 및 배포

#### Inception 설치

#### GOOGLE(GCP) BOSH 환경 설정

```
bosh create-env bosh.yml \
  --state=gcp/state.json \           # BOSH 설치 시 생성되는 파일로 절대 삭제 되면 안됨 (backup 필요)
  --vars-store gcp/creds.yml \       # BOSH 내부 인증서 파일 중요 (Backup필요)
  -o gcp/cpi.yml \                   # google CPI 적용
  -o uaa.yml \
  -o credhub.yml \
  -o jumpbox-user.yml \
  -v inception_os_user_name='ubuntu' \
  -v internal_cidr=192.168.10.0/24 \  # internal ip range
  -v internal_gw=192.168.10.1 \      # internal ip gateway
  -v internal_ip=192.168.10.6 \      # internal ip
  -v director_name='micro-bosh' \    # BOSH director 명
  -v network='paasta-v50-vpc' \      # google Network Name
  -v subnetwork='paasta-v50-private-subnet-1' \ # google Subnet Name
  -v tags=[paasta-v50-security] \    # google tag
  -v project_id=paas-ta-198701 \     # google Project Id
  -v private_key=~/.ssh/vcap.pem \   # ssh private key path
  -v zone=asia-northeast1-a \        # google zone
  --var-file gcp_credentials_json=~/.ssh/PaaS-TA-1e47e2554132.json # google service account key
```

## Director 설치

### » 샘플환경 설명 및 배포

#### Inception 설치

#### BOSH-LITE 환경 설정

```
bosh create-env bosh.yml \
  --state=warden/state.json \    # BOSH 설치 시 생성되는 파일로 절대 삭제 되면 안됨 (backup 필요)
  --vars-store warden/creds.yml \    # BOSH 내부 인증서 파일 중요 (Backup필요)
  -o virtualbox/cpi.yml \          # virtualbox CPI 적용
  -o virtualbox/outbound-network.yml \
  -o bosh-lite.yml \
  -o bosh-lite-runc.yml \
  -o uaa.yml \
  -o credhub.yml \
  -o jumpbox-user.yml \
  -v inception_os_user_name='ubuntu' \
  -v director_name='micro-bosh' \
  -v internal_ip=192.168.150.4 \      # internal ip range
  -v internal_gw=192.168.150.1 \    # internal gateway
  -v internal_cidr=192.168.150.0/24 \ # internal ip rang
  -v network_name=vboxnet0 \        # network name
  -v outbound_network_name=NatNetwork # outbound network
```

## Director 설치

### » 샘플환경 설명 및 배포

#### Inception 설치

#### VirtualBox 설치

#### VirtualBox 설치

```
$ sudo apt update
$ sudo apt upgrade
$ wget -q https://www.virtualbox.org/download/oracle_vbox_2016.asc -O- | sudo apt-key add -
$ wget -q https://www.virtualbox.org/download/oracle_vbox.asc -O- | sudo apt-key add -
$ sudo add-apt-repository "deb http://download.virtualbox.org/virtualbox/debian bionic contrib"
$ sudo apt update
$ sudo apt install virtualbox-6.0
$ VBoxManage --version
6.0.14r133895
```

## 03

## Director 설치

## » 샘플환경 설명 및 배포

## Inception 설치

## BOSH Deploy

```
$ cd ~/workspace/paasta-5.0/deployment/bosh-deployment  
$ ./deploy-{IaaS}.sh
```

## BOSH 설치 실행 화면

```
ubuntu@ubuntu-15zd90n-vx7bk:~/workspace/paasta-5.0/deployment/bosh-deployment$ ./deploy-bosh-lite.sh  
Deployment manifest: '/home/ubuntu/workspace/paasta-5.0/deployment/bosh-deployment/bosh.yml'  
Deployment state: 'warden/state.json'  
  
Started validating  
Validating release 'bosh'... Finished (00:00:00)  
Validating release 'bpm'... Finished (00:00:00)  
Validating release 'bosh-virtualbox-cpi'... Finished (00:00:02)  
Validating release 'garden-runc'... Finished (00:00:01)  
Validating release 'bosh-warden-cpi'... Finished (00:00:00)  
Validating release 'os-conf'... Finished (00:00:00)  
Validating release 'uaa'... Finished (00:00:00)  
Validating release 'credhub'... Finished (00:00:00)  
Validating cpi release... Finished (00:00:00)
```

## 03

## Director 설치

## » 샘플환경 설명 및 배포

Inception 설치

BOSH Deploy

BOSH 설치 완료 화면

```
Updating instance 'bosh/0'... Finished (00:01:02)
Waiting for instance 'bosh/0' to be running... Finished (00:00:55)
Running the post-start scripts 'bosh/0'... Finished (00:00:01)
Finished deploying (00:03:02)

Cleaning up rendered CPI jobs... Finished (00:00:00)

Succeeded
ubuntu@ubuntu-15zd90n-vx7bk:~/workspace/paasta-5.0/deployment/bosh-deployment$
```

## 03

## Director 설치

## » 샘플환경 설명 및 배포

## Inception 설치

## BOSH Login

BOSH가 설치 되면 BOSH설치 디렉토리 이하 `{laaS}/creds.yml`이 생성됨

`creds.yml`은 **BOSH 인증정보**를 가지고 있으며 `creds.yml`을 활용해 BOSH에 login

BOSH 로그인 후 `bosh-cli` 명령어를 이용하여 **PaaS-TA를 설치**할 수 있음

```
$ cd ~/workspace/paasta-5.0/deployment/bosh-deployment
$ export BOSH_CA_CERT=$(bosh int ./laaS/creds.yml --path /director_ssl/ca)
$ export BOSH_CLIENT=admin
$ export BOSH_CLIENT_SECRET=$(bosh int ./laaS/creds.yml --path /admin_password)
$ bosh alias-env {director_alias} -e {bosh-internal-ip} --ca-cert <(bosh
int ./laaS/creds.yml --path /director_ssl/ca)
$ bosh -e {director_alias} env
```



## 03

## Director 설치

## » 샘플환경 설명 및 배포

## Inception 설치

## jumpbox

BOSH설치 시 option file에 **jumpbox-user.yml**을 추가

Jumpbox는 **BOSH VM에 접근하기 위한 인증을 적용**하게 됨

인증 key는 **BOSH 자체적으로 생성**하며 **인증키를 통해 BOSH VM에 접근**할 수 있음

```
$ cd ~/workspace/paasta-5.0/deployment/bosh-deployment
$ bosh int ./[laaS]/creds.yml --path /jumpbox_ssh/private_key > jumpbox.key
$ chmod 600 jumpbox.key
$ ssh jumpbox@[bosh-internal-ip] -i jumpbox.key
```

## 03

## Director 설치

## » 샘플환경 설명 및 배포

## Inception 설치

## jumpbox

```
ubuntu@ubuntu-15zd90n-vx7bk:~/workspace/paasta-5.0/deployment/bosh-deployment$ ssh jumpbox@10.0.1.6 -i jumpbox.key
The authenticity of host '10.0.1.6 (10.0.1.6)' can't be established.
ECDSA key fingerprint is SHA256:vWe8k5Ts/+09Wkqc5SHTXH/YFw0VIF27sxWl+Tq0x/I.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '10.0.1.6' (ECDSA) to the list of known hosts.
Unauthorized use is strictly prohibited. All access and activity
is subject to logging and monitoring.
Welcome to Ubuntu 16.04.6 LTS (GNU/Linux 4.15.0-54-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

Last login: Thu Mar 26 07:18:36 2020 from 10.0.1.1
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

bosh/0:~$
```

## 03

## Director 설치

## » 샘플환경 설명 및 배포

## Inception 설치

## credhub

BOSH설치 시 option file에 **credhub.yml**을 추가

credhub은 인증정보 저장소

BOSH 설치 시 credhub.yml을 적용하면 PaaS-TA 설치 시  
**인증정보를 credhub에 저장**하게 됨

credhub에 로그인 하기 위해서는 credhub cli를 통해  
인증정보를 조회, 수정 및 삭제 할 수 있음

## 03

## Director 설치

## » 샘플환경 설명 및 배포

## Inception 설치

## credhub

credhub cli install

```
$ wget https://github.com/cloudfoundry-incubator/credhub-  
cli/releases/download/2.0.0/credhub-linux-2.0.0.tgz  
$ tar -xvf credhub-linux-2.0.0.tgz  
$ chmod +x credhub  
$ sudo mv credhub /usr/local/bin/credhub  
$ credhub --version
```

## 03

## Director 설치

## » 샘플환경 설명 및 배포

## Inception 설치

## credhub

credhub login

```
$ export CREDHUB_CLIENT=credhub-admin  
$ export CREDHUB_SECRET=$(bosh int --path /credhub_admin_client_secret ./{IaaS}/creds.yml)  
$ export CREDHUB_CA_CERT=$(bosh int --path /credhub_tls/ca ./{IaaS}/creds.yml)  
$ credhub login -s https://{bosh-internal-ip}:8844 --skip-tls-validation  
$ credhub find
```

## 03

## Director 설치

## » 샘플환경 설명 및 배포

## Inception 설치

## credhub

credhub login 후 find를 해보면 비어 있는 것을 알 수 있음

PaaS-TA를 설치하면 인증 정보가 저장되어 조회 할 수 있음

```
$ credhub find
```

ex) uaa 인증정보 조회

```
$ credhub get -n /{director}/{deployment}/uaa_ca
```

## 04 Bosh를 통한 PaaS-TA 구축

### » cloud-config

### PaaS-TA 설치를 위해 IaaS 관련 network, storage, vm 설정 정의

IaaS, network, disk 등 상황에 따라 설정이 다름

paasta-deployment.yml은 cloud-config 설정에 따라 paasta-vm을 설치

PaaS-TA Deploy 전에 cloud-config가 BOSH에 Upload 되어야 함

PaaS-TA는 IaaS별 cloud-config 예제를 제공하며,  
PaaS-TA를 설치 하려면 cloud-config.yml을 IaaS상황에 맞게 수정해야 함

## 04 Bosh를 통한 PaaS-TA 구축

### » cloud-config

openstack 기준 cloud-config 설정 예제

paasta-5.0 deployment를 다운로드 받으면 cloud-config 디렉토리 이하에  
IaaS별 cloud-config 예제를 볼 수 있음

```
ubuntu@ubuntu-15zd90n-vx7bk:~/workspace/paasta-5.0/deployment/cloud-config$ ls
aws-cloud-config.yml      bosh-lite-cloud-config.yml  openstack-cloud-config.yml
azure-cloud-config.yml    gcp-cloud-config.yml        vsphere-cloud-config.yml
```



## 04 Bosh를 통한 PaaS-TA 구축

### » cloud-config

#### openstack 기준 cloud-config 설정 예제

## azs는 가용 zone에 대한 설정으로 paasta가 설치될 zone에 대한 정의이다.

azs:

- name: z1
  - cloud\_properties:
    - availability\_zone: nova
- name: z2
  - cloud\_properties:
    - availability\_zone: nova
- name: z3
  - cloud\_properties:
    - availability\_zone: nova
- name: z4
  - cloud\_properties:
    - availability\_zone: nova
- name: z5
  - cloud\_properties:
    - availability\_zone: nova
- name: z6
  - cloud\_properties:
    - availability\_zone: nova

# 04 Bosh를 통한 PaaS-TA 구축

## » cloud-config

### openstack 기준 cloud-config 설정 예제

```
## vm_type은 paasta vm이 설치될 spec을 정의하며 instance_type은 openstack flavor에 설정 되어 있어야 한다.
vm_types:
  - name: minimal
    cloud_properties:
      instance_type: m1.tiny
  - name: default
    cloud_properties:
      instance_type: m1.medium
  - name: small
    cloud_properties:
      instance_type: m1.small
  - name: medium
    cloud_properties:
      instance_type: m1.medium
  - name: medium-memory-8GB
    cloud_properties:
      instance_type: m1.medium
  - name: large
    cloud_properties:
      instance_type: m1.large
  - name: xlarge
    cloud_properties:
      instance_type: m1.xlarge
  - name: small-50GB
    cloud_properties:
      instance_type: m1.medium
  - name: small-50GB-ephemeral-disk
    cloud_properties:
      instance_type: m1.medium
  - name: small-100GB-ephemeral-disk
    cloud_properties:
      instance_type: m1.large
  - name: small-highmem-100GB-ephemeral-disk
    cloud_properties:
      instance_type: m1.large
  - name: small-highmem-16GB
    cloud_properties:
      instance_type: m1.large-memory
  - name: service_medium
    cloud_properties:
      instance_type: m1.medium
  - name: service_medium_2G
    cloud_properties:
      instance_type: m1.medium
  - name: portal_small
    cloud_properties:
      instance_type: m1.tiny
  - name: portal_medium
    cloud_properties:
      instance_type: m1.small
  - name: portal_large
    cloud_properties:
      instance_type: m1.small
```

## 04 Bosh를 통한 PaaS-TA 구축

### » cloud-config

#### openstack 기준 cloud-config 설정 예제

## compilation은 PaaS-TA 설치 시 Compile VM이 생성될 zone 및 vm type을 설정한다.

compilation:

az: z3

network: default

reuse\_compilation\_vms: true

vm\_type: large

workers: 5

## compilation은 PaaS-TA 설치 시 Compile VM이 생성될 zone 및 vm type을 설정한다.

disk_types:	- disk_size: 5120	name: 30GB
- disk_size: 1024	name: 5GB	- disk_size: 51200
name: default	- disk_size: 8192	name: 50GB
- disk_size: 1024	name: 8GB	- disk_size: 102400
name: 1GB	- disk_size: 10240	name: 100GB
- disk_size: 2048	name: 10GB	- disk_size: 1048576
name: 2GB	- disk_size: 20480	name: 1TB
- disk_size: 4096	name: 20GB	
name: 4GB	- disk_size: 30720	

## 04 Bosh를 통한 PaaS-TA 구축

### » cloud-config

#### openstack 기준 cloud-config 설정 예제

- cloud\_properties:
  - type: SSD1
  - disk\_size: 2000
  - name: 2GB\_GP2
- cloud\_properties:
  - type: SSD1
  - disk\_size: 5000
  - name: 5GB\_GP2
- cloud\_properties:
  - type: SSD1
  - disk\_size: 10000
  - name: 10GB\_GP2
- cloud\_properties:
  - type: SSD1
  - disk\_size: 50000
  - name: 50GB\_GP2

## 04 Bosh를 통한 PaaS-TA 구축

### » cloud-config

#### openstack 기준 cloud-config 설정 예제

```
## network은 paasta vm이 생성될 network관련 정의이다.
   openstack의 subnet및 security_group, dns, gateway 등에 대한 정보를 설정한다.
networks:
- name: default
  subnets:
  - az: z1
    cloud_properties:
      name: random
      net_id: 51b96a68-aded-4e73-aa44-f44a812b9b30
      security_groups:
      - paasta-v50-security
    dns:
    - 8.8.8.8
    gateway: 10.20.10.1
    range: 10.20.10.0/24
    reserved:
    - 10.20.10.2 - 10.20.10.10
    static:
    - 10.20.10.11 - 10.20.10.30
```

# 04 Bosh를 통한 PaaS-TA 구축

## » cloud-config

### openstack 기준 cloud-config 설정 예제

```
- az: z2
  cloud_properties:
    name: random
    net_id: 51b96a68-aded-4e73-aa44-f44a812b9b30
    security_groups:
      - paasta-v50-security
  dns:
    - 8.8.8.8
  gateway: 10.20.20.1
  range: 10.20.20.0/24
  reserved:
    - 10.20.20.2 - 10.20.20.10
  static:
    - 10.20.20.11 - 10.20.20.30
- az: z3
  cloud_properties:
    name: random
    net_id: 51b96a68-aded-4e73-aa44-f44a812b9b30
    security_groups:
      - paasta-v50-security
  dns:
    - 8.8.8.8
  gateway: 10.20.30.1
  range: 10.20.30.0/24
  reserved:
    - 10.20.30.2 - 10.20.30.10
  static:
    - 10.20.30.11 - 10.20.30.30
```

```
- az: z3
  cloud_properties:
    name: random
    net_id: 51b96a68-aded-4e73-aa44-f44a812b9b30
    security_groups:
      - paasta-v50-security
  dns:
    - 8.8.8.8
  gateway: 10.20.30.1
  range: 10.20.30.0/24
  reserved:
    - 10.20.30.2 - 10.20.30.10
  static:
    - 10.20.30.11 - 10.20.30.30
- az: z4
  cloud_properties:
    name: random
    net_id: 51b96a68-aded-4e73-aa44-f44a812b9b30
    security_groups:
      - paasta-v50-security
  dns:
    - 8.8.8.8
  gateway: 10.20.40.1
  range: 10.20.40.0/24
  reserved:
    - 10.20.40.2 - 10.20.40.10
  static:
    - 10.20.40.11 - 10.20.40.30
```

## 04 Bosh를 통한 PaaS-TA 구축

### » cloud-config

#### openstack 기준 cloud-config 설정 예제

```

- name: vip
  type: vip

- name: service_private
  subnets:
  - az: z5
    cloud_properties:
      name: random
      net_id: 51b96a68-aded-4e73-aa44-f44a812b9b30
      security_groups:
        - paasta-v50-security
    dns:
      - 8.8.8.8
    gateway: 10.20.50.1
    range: 10.20.50.0/24
    reserved:
      - 10.20.50.2 - 10.20.50.10
    static:
      - 10.20.50.11 - 10.20.50.30

- az: z6
  cloud_properties:
    name: random
    net_id: 51b96a68-aded-4e73-aa44-f44a812b9b30
    security_groups:
      - paasta-v50-security
    dns:
      - 8.8.8.8
    gateway: 10.20.60.1
    range: 10.20.60.0/24
    reserved:
      - 10.20.60.2 - 10.20.60.10
    static:
      - 10.20.60.11 - 10.20.60.30

- name: service_public
  type: vip

```

## 04 Bosh를 통한 PaaS-TA 구축

### » cloud-config

#### openstack 기준 cloud-config 설정 예제

## vm\_extensions는 관련 보안 그룹 및 로드 밸런서와 같은 임의의 IaaS 특정 구성을 지정할 수 있는 클라우드 구성의 가상 컴퓨터 구성이다.

vm\_extensions:

- cloud\_properties:

- ports:

- host: 3306

- name: mysql-proxy-lb

- name: cf-router-network-properties

- name: cf-tcp-router-network-properties

- name: diego-ssh-proxy-network-properties

- name: cf-haproxy-network-properties

- cloud\_properties:

- ephemeral\_disk:

- size: 51200

- type: gp2

- name: small-50GB

- cloud\_properties:

- ephemeral\_disk:

- size: 102400

- type: gp2

- name: small-highmem-100GB



## 04 Bosh를 통한 PaaS-TA 구축

### » cloud-config

openstack 기준 cloud-config 설정 예제

cloud-config 설정 정보를 BOSH에 update

```
$ cd ~/workspace/paasta-5.0/deployment/cloud-config  
$ bosh -e {director_alias} update-cloud-config {IaaS}-cloud-config.yml
```

BOSH에 upload된 cloud-config 확인

```
$ bosh -e {director_alias} cloud-config
```

# 04 Bosh를 통한 PaaS-TA 구축

## » cloud-config

### openstack 기준 cloud-config 설정 예제

다음은 openstack에서 정의한 flavor type 예제

<input type="checkbox"/> Flavor Name	VCPUs	RAM	Root Disk	Ephemeral Disk	Swap Disk	RX/TX factor	ID	Public	Metadata	Actions
<input type="checkbox"/> inception	1	4GB	80GB	0GB	0MB	1.0	2c09f880-0b33-4168-bad8-75e606ea22b0	Yes	No	<a href="#">Edit Flavor</a> ▼
<input type="checkbox"/> kubo	3	8GB	40GB	0GB	0MB	1.0	6	Yes	No	<a href="#">Edit Flavor</a> ▼
<input type="checkbox"/> m1.large	4	8GB	100GB	0GB	0MB	1.0	f84e4b50-dbd9-4e8b-83fc-4c0f4a4f88f1	No	No	<a href="#">Edit Flavor</a> ▼
<input type="checkbox"/> m1.large-memory	4	16GB	50GB	0GB	0MB	1.0	25bfe587-87eb-401b-9a8f-9d683e801d49	Yes	No	<a href="#">Edit Flavor</a> ▼
<input type="checkbox"/> m1.medium	2	4GB	50GB	0GB	0MB	1.0	3	Yes	No	<a href="#">Edit Flavor</a> ▼
<input type="checkbox"/> m1.medium-memory	2	8GB	40GB	0GB	0MB	1.0	4f6813c3-f486-411a-b3d5-3f90bd0277e9	Yes	No	<a href="#">Edit Flavor</a> ▼
<input type="checkbox"/> m1.monitoring	1	4GB	30GB	0GB	0MB	1.0	7f076c9f-7204-4a6c-971f-843931cb797e	Yes	No	<a href="#">Edit Flavor</a> ▼
<input type="checkbox"/> m1.small	1	2GB	30GB	0GB	0MB	1.0	2	Yes	No	<a href="#">Edit Flavor</a> ▼
<input type="checkbox"/> m1.tiny	1	512MB	10GB	0GB	0MB	1.0	1	Yes	No	<a href="#">Edit Flavor</a> ▼
<input type="checkbox"/> m1.xlarge	8	16GB	160GB	0GB	0MB	1.0	5	Yes	No	<a href="#">Edit Flavor</a> ▼

## 04 Bosh를 통한 PaaS-TA 구축

### » runtime-config

#### runtime-config 설정 예제

#### runtime-config 설정 정보를 BOSH에 update

```
$ cd ~/workspace/paasta-5.0/deployment/bosh-deployment
$ chmod 755 ./update-runtime-config.sh
$ cat ./update-runtime-config.sh
bosh -e ${director_alias} update-runtime-config -n runtime-configs/dns.yml \
  -v inception_os_user_name=${inception_os_user_name} \
  -v cert_days=3650
$ ./update-runtime-config.sh
```

#### BOSH에 upload된 runtime-config 확인

```
$ bosh -e {director_alias} runtime-config
```

## 04 Bosh를 통한 PaaS-TA 구축

### » manifest 파일 분석 및 배포 예시

paasta deploy shell

paasta-deployment.yml 파일은 PaaS-TA를 deploy하는 manifest file

paasta vm에 대한 설치를 정의

vm중 singleton-blobstore, database의 azs(zone)을 변경 하면  
Org & Space 및 app의 정보가 모두 삭제됨

## 04 Bosh를 통한 PaaS-TA 구축

### » [교육장 - virtualbox]

#### Virtualbox 기준 route 설정 예제

```
# Mac OS X
$ sudo route add -net 10.244.0.0/16 192.168.50.6

# Linux (using iproute2 suite)
$ sudo ip route add 10.244.0.0/16 via 192.168.50.##

# Windows
$ route add 10.244.0.0/16 192.168.50.6
```

#### route 확인

```
# Linux
$ ip route list
```

## 04 Bosh를 통한 PaaS-TA 구축

### » manifest 파일 분석 및 배포 예시

#### paasta deploy shell

아래 option들은 존재 하지 않으면 BOSH가 자동 생성해줌

cf_admin_password	paasta admin password
cc_db_encryption_key	paasta database 암호화 키 (version upgrade시 동일한 key이어야 함)
uaa_database_password	uaadb database admin pwd
cc_database_password	cloud_controller(api) database admin pwd
cert_days	paasta deploy 시 내부 컴포넌트끼리 통신하기 위한 인증서를 사용 인증서는 Bosh(credhub)에서 생성하며 인증서 유효 기간을 원하는 기간을 정의 (일단위)
uaa_login_logout_redirect_parameter_whitelist	uaa whitelist 추가 http://portal-web-user.[User Portal url].xip.io 및 http://portal-web-user.[User Portal url].xip.io/callback 입력
uaa_login_branding_company_name	uaa 페이지 타이틀 명
uaa_login_branding_footer_legal_text	uaa 페이지 하단영역 텍스트
uaa_login_branding_product_logo	uaa 페이지 로고 이미지(base64)
uaa_login_branding_square_logo	uaa 페이지 타이틀 로고 이미지(base64)
uaa_login_links_passwd	# uaa 페이지에서 reset password 누를시 이동하는 영역 설정 http://portal-web-user.[User Portal url].xip.io/resetpaawd
uaa_login_links_signup	uaa 페이지에서 create account 누를시 이동하는 영역 설정 http://portal-web-user.[User Portal url].xip.io/createuser
uaa_clients_portalclient_redirect_uri	uaa 페이지에서 redirect 영역 설정 http://portal-web-user.[User Portal url].xip.io 및 http://portal-web-user.[User Portal url].xip.io/callback 입력

 Note : User Portal url은 앞으로 Portal에 설치할 haproxy의 public ip

# 04 Bosh를 통한 PaaS-TA 구축

## » manifest 파일 분석 및 배포 예시

### deploy-openstack.sh

```

bosh -e {director_alias} -d paasta deploy paasta-deployment.yml \    # paasta manifest file
-o operations/openstack.yml \    # openstack 설정
-o operations/use-compiled-releases.yml \    # compile된 release 파일 정보(offline)
-o operations/use-haproxy.yml \
-o operations/use-haproxy-public-network.yml \
-o operations/use-postgres.yml \    # database postgres 선택
-v inception_os_user_name=ubuntu \    # home user명 (release file path와 연관성 있음. /home/ubuntu/paasta-5.0 이하 release 파일들의 경로 설정)
-v haproxy_public_ip=52.199.190.1 \    # paasta public ip
-v haproxy_public_network_name=vip \
-v cf_admin_password=admin \    # paasta admin password
-v cc_db_encryption_key=db-encryption-key \    # version upgrade시 동일한 key이어야 함
-v uaa_database_password=uaa_admin \    # uadb database pwd
-v cc_database_password=cc_admin \    # ccdb database pwd
-v cert_days=3650 \    # paasta 인증서 유효기간
-v system_domain=52.199.190.1.xip.io \    # domain (xip.io를 사용하는 경우 haproxy public_ip와 동일하게 함)
-v uaa_login_logout_redirect_parameter_disable=false \
-v uaa_login_logout_redirect_parameter_whitelist=[http://portal-web-user.115.68.46.190.xip.io,"http://portal-web-
user.115.68.46.190.xip.io/callback","http://portal-web-user.115.68.46.190.xip.io/login"] \    # uaa whitelist 추가 http://portal-web-user.[User Portal
url].xip.io 및 http://portal-web-user.[User Portal url].xip.io/callback 입력(User Portal url은 앞으로 포탈에 설치할 haproxy의 public ip)
-v uaa_login_branding_company_name="PaaS-TA R&D" \    # uaa 페이지 타이틀 명
-v uaa_login_branding_footer_legal_text="Copyright © PaaS-TA R&D Foundation, Inc. 2017. All Rights Reserved." \    # uaa 페이지 하단영역 텍스트
-v uaa_login_branding_product_logo="iVBORw0KGgoAAA...AAAAASUVORK5CYII=" \    # uaa 페이지 로고 이미지(base64)
-v uaa_login_branding_square_logo="iVBORw0KGgoAAA...AAAAASUVORK5CYII=" \    # uaa 페이지 타이틀 로고 이미지(base64)
-v uaa_login_links_passwd="http://portal-web-user.115.68.46.190.xip.io/resetpasswd" \    # uaa 페이지에서 reset password 누를 시 이동하는 영역
설정 http://portal-web-user.[User Portal url].xip.io/resetpasswd (User Portal url은 앞으로 포탈에 설치할 haproxy의 public ip)
-v uaa_login_links_signup="http://portal-web-user.115.68.46.190.xip.io/createuser" \    # uaa 페이지에서 create account 누를 시 이동하는 영역
설정 http://portal-web-user.[User Portal url].xip.io/createuser (User Portal url은 앞으로 포탈에 설치할 haproxy의 public ip)
-v uaa_clients_portalclient_redirect_uri="http://portal-web-user.115.68.46.190.xip.io,http://portal-web-user.115.68.46.190.xip.io/callback" \ # uaa
페이지에서 redirect 영역 설정 http://portal-web-user.[User Portal url].xip.io 및 http://portal-web-user.[User Portal url].xip.io/callback 입력 (User Portal
url은 앞으로 포탈에 설치할 haproxy의 public ip)
-v uaa_clients_portalclient_secret="portalclient" \    # uaa portalclient secret 설정
-v uaa_admin_client_secret="admin-secret" \    # uaac admin계정 secret 설정

```

로고 이미지를 바이너리화하여  
문자로 표현한 형식을 축약하여 표시

## 04 Bosh를 통한 PaaS-TA 구축

### » PaaS-TA Deploy 실행

```
$ ./deploy-bosh-lite.sh
```

```
ubuntu@ubuntu-15zd90n-vx7bk:~/workspace/paasta-5.0/deployment/paasta-deployment$ ./deploy-bosh-lite.sh
Using environment '10.0.1.6' as client 'admin'

Using deployment 'paasta'

##### 100.00% 164.84 MiB/s 0s
Task 3

Task 3 | 08:23:38 | Extracting release: Extracting release (00:00:00)
Task 3 | 08:23:38 | Verifying manifest: Verifying manifest (00:00:00)
Task 3 | 08:23:38 | Resolving package dependencies: Resolving package dependencies (00:00:00)
Task 3 | 08:23:38 | Creating new packages: binary-buildpack-cflinuxfs2/952b357e6b8fe115be77e95424261eab-
Task 3 | 08:23:38 | Creating new packages: binary-buildpack-cflinuxfs3/a406ca0cffa681999ba325905063ef2b-
Task 3 | 08:23:38 | Creating new packages: binary-buildpack-windows/c0aecbbd98944961a57f53692b1927c11f7-
Task 3 | 08:23:38 | Creating new packages: binary-buildpack-windows2012R2/6baa8a9d28534404ea114182e9e33-
Task 3 | 08:23:38 | Creating new packages: binary-buildpack-windows2016/5dc0d08fa57b3fd52a49fb144fcb147-
Task 3 | 08:23:38 | Creating new compiled packages: binary-buildpack-cflinuxfs2/952b357e6b8fe115be77e95-

Task 35 Started Thu Mar 26 08:25:46 UTC 2020
Task 35 Finished Thu Mar 26 08:43:28 UTC 2020
Task 35 Duration 00:17:42
Task 35 done

Succeeded
ubuntu@ubuntu-15zd90n-vx7bk:~/workspace/paasta-5.0/deployment/paasta-deployment$
```



# 04 Bosh를 통한 PaaS-TA 구축

## » PaaS-TA Deploy VM 확인

```
$ bosh -e {director_alias} -d {deploy_name} vms
```

```
ubuntu@ubuntu-15zd90n-vx7bk:~/workspace/paasta-5.0/deployment/paasta-deployment$ bosh -e micro-bosh -d paasta vms
Using environment '10.0.1.6' as client 'admin'
```

Task 36. Done

Deployment 'paasta'

Instance	Process State	AZ	IPs	VM CID	VM Type	Active
adapter/c549b31b-d683-4c88-a10a-39e4a943198f	running	z1	10.244.0.129	c2c113e9-fe11-4ecb-435b-acfb73ba0975	minimal	true
api/20a7c11c-507d-4b47-ae23-ce6948904550	running	z1	10.244.0.134	807fc0c5-bf99-4f2c-5249-1cebc9f06979	small	true
cc-worker/c2dc47d5-f695-4694-9fbc-35d4fe46cb8d	running	z1	10.244.0.135	1c5762c8-4292-4d36-542d-97ae4d9a0335	minimal	true
credhub/a04c66a5-cea1-423c-9262-bbf352041730	running	z1	10.244.0.141	81fd4421-ac85-48d6-6031-476064dd7a86	minimal	true
database/c773e813-2914-4270-98db-7d28b1604c8c	running	z1	10.244.0.130	2565e4e2-cff7-470a-4287-3fca3e37a181	small	true
diego-api/5c34f1dc-58f6-413e-b7e0-b5b8972e1dc8	running	z1	10.244.0.131	429aaaa1-12a7-48a3-67b7-549475a18b39	small	true
diego-cell/2674e655-21ea-4abd-8bc5-69ba21835de8	running	z1	10.244.0.139	96f53dc4-f848-492e-68bf-13c62e2ddce1	small-highmem-16GB	true
doppler/5c40e15d-c027-464b-b149-edc4e763882f	running	z1	10.244.0.138	0bd2b2b4-bb12-49f2-56e6-faf0033b2d6b	minimal	true
log-api/e3412984-1306-4536-b7cd-2178102a5acb	running	z1	10.244.0.140	8f206dd2-4c95-4d3c-667f-103081d8826c	minimal	true
nats/4d939d4b-656e-4369-92f5-cc4ccd520137	running	z1	10.244.0.128	9889bf73-0051-4048-6cc8-b0f621bec020	minimal	true
router/26b79586-04bc-46f4-a859-8a0e457488d4	running	z1	10.244.0.34	a0bdc319-9f80-47ed-6fa4-0ccc884569b5	minimal	true
scheduler/80e58940-c8d4-49a3-93be-4c12e2d50bce	running	z1	10.244.0.136	3b1a8750-0e7a-4f05-4407-cb9fac48371e	minimal	true
singleton/blobstore/35978a2a-7b78-48c7-9564-a54c3149d9dd	running	z1	10.244.0.133	482f9459-7b41-46da-4ae3-16393d17d4d6	small	true
tcp-router/fec0689a-2f92-4611-a552-8c23a0fc4d43	running	z1	10.244.0.137	d0a8ba53-7222-4ca2-6621-5d718fd7547f	minimal	true
uaa/644bcef6-1c42-4056-bd0c-a3d963710241	running	z1	10.244.0.132	acfe86b8-2a83-499c-4bf4-3eb76b53d9f2	minimal	true

15 vms

Succeeded



MEMO





End of Document

